



Computation of Greeks using Malliavin's calculus in jump type market models

Marie-Pierre Bavouzet, Marouen Messaoud

► To cite this version:

Marie-Pierre Bavouzet, Marouen Messaoud. Computation of Greeks using Malliavin's calculus in jump type market models. [Research Report] RR-5482, INRIA. 2005, pp.31. inria-00070525

HAL Id: inria-00070525

<https://inria.hal.science/inria-00070525>

Submitted on 19 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Computation of Greeks using Malliavin's calculus in jump type market models

Marie-Pierre Bavouzet and Marouen Messaoud

N° 5482

February, 1 2005

_____ Thème NUM _____



*rapport
de recherche*



Computation of Greeks using Malliavin's calculus in jump type market models

Marie-Pierre Bavouzet* and Marouen Messaoud†

Thème NUM — Systèmes numériques
Projet MATHFI

Rapport de recherche n° 5482 — February, 1 2005 — 31 pages

Abstract: We use the Malliavin calculus for Poisson processes in order to compute sensitivities for European options with underlying following a jump type diffusion.

The main point is to settle an integration by parts formula (similar to the one in the Malliavin calculus) for a general multidimensional random variable which has an absolutely continuous law with differentiable density. We give an explicit expression of the differential operators involved in this formula and this permits to simulate them and consequently to run a Monte Carlo algorithm.

Key-words: Malliavin calculus, Monte-Carlo algorithm, Euler scheme, compound Poisson process, sensitivities computation.

* INRIA-Rocquencourt, email: marie-pierre.bavouzet@inria.fr

† INRIA-Rocquencourt, email: marouen.messaoud@inria.fr

Calcul des sensibilités via le calcul de Malliavin dans les marchés à sauts

Résumé : On utilise le calcul de Malliavin pour des processus de Poisson afin de calculer les sensibilités d'options européennes dont le sous-jacent suit une diffusion à sauts.

Le point principal est d'établir une formule d'intégration par parties (semblable à celle du calcul de Malliavin "standard") pour des variables aléatoires multidimensionnelles absolument continues par rapport à la mesure de Lebesgue, et dont la densité est différentiable. Nous donnons une expression explicite de chaque opérateur apparaissant dans cette formule. Cela nous permet alors de les simuler et donc de pouvoir mettre en oeuvre un algorithme de Monte Carlo.

Mots-clés : calcul de Malliavin, algorithme de Monte-Carlo, schéma d'Euler, processus de Poisson composé, calcul des sensibilités.

1 Introduction

In the last years, following the pioneering papers [9], [8] a lot of work concerning the numerical applications of the stochastic variational calculus (Malliavin calculus) has been done. This mainly concerns applications in mathematical finance: computations of conditional expectations (which appear in the american option pricing, for example) and of sensitivities (the so called Greeks). The models at hand are usually log-normal type diffusions and then one may use the standard Malliavin calculus. But nowadays people are more and more interested in jump type diffusions (see [5]) and then one has to use the stochastic variational calculus corresponding to Poisson point processes. Such a calculus has already been developed in [2] concerning the noise coming from the amplitude of the jumps and in [4] concerning the jump times. More recently in [3] one gives a unified approach using the language of the Dirichlet forms. Another point of view, based on chaos decomposition may be found in [7], [1], [12], [14], [11].

The aim of our paper is to give a concrete application of the Malliavin calculus approach for sensitivities computation (Greeks) for jump diffusion models. We give two examples: in the first one, we use the Malliavin calculus with respect to the jump amplitudes. In the second one, we add a Brownian part and differentiate with respect to both the jump amplitudes and to the Wiener increments.

The Malliavin integration by parts formula used in this paper is an elementary one. Notice that any numerical scheme used in a Monte Carlo algorithm appears as a function of a finite number of random variables H_1, \dots, H_n . It turns out that, if the law of these random variables is absolutely continuous with respect to the Lebesgue measure and has a smooth density, then the strategy developed in the Malliavin calculus may be implemented for H_1, \dots, H_n and an integration by parts formula is obtained. This is not specific for the Brownian motion or for a Poisson point process but represents an elementary abstract calculus.

The paper is organized as follows. In Section 1 we present the problem: we give the stochastic equation verified by the underlying asset which is a one dimensional jump diffusion (one may consider multi-dimensional diffusions as well but we keep in the one dimensional setting just to avoid technical difficul-

ties). Then we precise how an integration by parts formula may be used in the Monte Carlo algorithm in order to compute sensitivities. In Section 2 we give the integration by parts formula. In Section 3 we use the Malliavin's calculus with respect to the amplitudes of the jumps in order to compute sensitivities. In section 4, we consider a Merton's model in which stochastic integrals driven by a Brownian motion and by a compound Poisson process appear. We have the choice of using the Malliavin's calculus with respect to the Brownian motion, to the amplitudes of the jumps or to both of them. We perform the three algorithms and we compare the empirical variance of each of them. It turns out that the most performant algorithm (with the smallest variance) is that based on both the Brownian motion and the amplitudes of the jumps. So the numerical experiments (there is no theoretical result) indicate that one has to use the integration by parts formula with respect to all the noise which is available in the model. The numerical results are contained in section 5. We compare the Malliavin method with the finite difference method. For European call options, the results are rather similar but for digital options, the results given by the Malliavin method are much better than the ones given by the finite difference method. This can be explained by the fact that the payoff function is much more irregular.

2 The problem

We compute the sensitivities of an option with payoff ϕ , where the asset $(S_t)_{t \geq 0}$ follows a one-dimensional jump diffusion process driven by a compound Poisson process. Let us precise our problem.

Let N be a compound Poisson process. We denote by $(T_i)_{i \in \mathbb{N}}$ the jump times of $t \rightarrow N_t$, and by $(J_t)_{t \geq 0}$ the corresponding standard Poisson process with parameter $\lambda > 0$, that is J_t counts the number of jumps up to t and for all $n \geq 1$, $T_n - T_{n-1}$ has exponential distribution with mean λ . For all $i \in \mathbb{N}^*$, we define $\Delta_i = N_{T_i} - N_{T_{i-1}}$, so Δ_i represents the jump amplitude at time T_i . The random variables $(\Delta_i)_{i \in \mathbb{N}^*}$ are independent and identically distributed. We assume that the law of Δ_i is absolutely continuous with respect to the Lebesgue measure and we denote by ρ its density, that is $\Delta_i \sim \rho(a) da$, for all $i \in \mathbb{N}^*$. We will suppose that ρ is differentiable and vanishes as $x \rightarrow \pm\infty$ (see section

4).

We deal with two models. In the first one, the asset S_t is a pure jump diffusion solution of the SDE

$$S_t = \beta + \sum_{i=1}^{J_t} c(T_i, \Delta_i, S_{T_i}^-) + \int_0^t b(u, S_u) du. \quad (1)$$

In the second one, we add a Brownian part :

$$S_t = \beta + \sum_{i=1}^{J_t} c(T_i, \Delta_i, S_{T_i}^-) + \int_0^t b(u, S_u) du + \int_0^t \sigma(u, S_u) dW_u. \quad (2)$$

Our aim is to compute $\partial_\beta E(\phi(S_T))$, that is the Delta of an European option of payoff ϕ . In several papers (such as [8], [9], [10]), one treat this problem using a Malliavin calculus approach. We will follow a similar strategy in our frame, using Malliavin calculus for jump type diffusions. We write

$$\begin{aligned} \frac{\partial}{\partial \beta} E(\phi(S_T)) &= E \left(\phi'(S_T) \frac{\partial S_T}{\partial \beta} \right) \\ &= E \left(\phi'(S_T) \frac{\partial S_T}{\partial \beta} \mathbf{1}_{J_T=0} \right) \\ &\quad + \sum_{n=1}^{\infty} E \left(E \left(\phi'(S_T) \frac{\partial S_T}{\partial \beta} \middle| \sigma(T_i, i \in \mathbb{N}) \right) \mathbf{1}_{J_T=n} \right). \end{aligned}$$

On $\{J_T = 0\}$ there is no jump on $]0, T]$, thus S_T and $\frac{\partial S_T}{\partial \beta}$ solve a deterministic integral equation. In the examples that we consider in this paper, the solution of these equations are explicit, so that this term is explicitly known.

Assume now that $J_T = n \neq 0$. Then, using an integration by parts formula on $\{J_T = n\}$, we write :

$$\begin{aligned} E \left(\phi'(S_T) \frac{\partial S_T}{\partial \beta} \middle| \sigma(T_i, i \in \mathbb{N}) \right) \mathbf{1}_{J_T=n} \\ = E \left(\phi(S_T) H_n(S_T, \frac{\partial S_T}{\partial \beta}) \middle| \sigma(T_i, i \in \mathbb{N}) \right) \mathbf{1}_{J_T=n}, \end{aligned}$$

where H_n is a weight involving “Malliavin derivatives” of S_T and of $\frac{\partial S_T}{\partial \beta}$. These differential operators are similar to those in [2], but the frame here is much more simple, since there are no accumulation of small jumps. So we will derive this integration by parts formula using elementary arguments. We obtain

$$\begin{aligned} \sum_{n=1}^{\infty} \mathbb{E} \left(\mathbb{E} \left(\phi'(S_T) \frac{\partial S_T}{\partial \beta} \mid \sigma(T_i, i \in \mathbb{N}) \right) \mathbf{1}_{J_T=n} \right) \\ = \mathbb{E} \left(\phi(S_T) H_{J_T}(S_T, \frac{\partial S_T}{\partial \beta}) \mathbf{1}_{J_T \geq 1} \right). \end{aligned}$$

In order to employ this formula in a Monte-Carlo algorithm, we proceed as follows. We simulate the jump times $(T_n^i)_{n \in \mathbb{N}, i = 1, \dots, M}$ and the jump amplitudes $(\Delta_n^i)_{n \in \mathbb{N}, i = 1, \dots, M}$, and we compute the corresponding J_T^i , S_T^i and $H_{J_T^i}^i$. Then we write

$$\mathbb{E} \left(\phi(S_T) H_{J_T}(S_T, \frac{\partial S_T}{\partial \beta}) \mathbf{1}_{J_T \geq 1} \right) \simeq \frac{1}{M} \sum_{i=1}^M \phi(S_T^i) H_{J_T^i}^i \mathbf{1}_{J_T^i \geq 1}.$$

In the case of Merton’s model we proceed as follows. First of all we simulate the jump times T_i , $i \in \mathbb{N}$. Once they are fixed we construct an Euler scheme in which the times T_i , $i \in \mathbb{N}$ are included in the discretization grid. Then we simulate the amplitudes of the jumps and the Brownian increments and we perform the Monte-Carlo algorithm.

3 Malliavin calculus for simple functionals

On a probability space (Ω, \mathcal{F}, P) we consider a sequence of independent random variables $(V_n, n \in \mathbb{N}^*)$. We suppose that

Hypothesis 3.1 *For every $n \in \mathbb{N}^*$, the law of V_n is absolutely continuous with respect to the Lebesgue measure and has the density p_n which is continuously differentiable on \mathbb{R} and such that $\forall k \in \mathbb{N} \lim_{y \rightarrow \pm\infty} |y|^k p(y) = 0$. We also assume*

that $\partial_y \ln(p(y)) = \frac{\partial_y p(y)}{p(y)}$ has at most polynomial growth.

We introduce some notation. For $k \geq 1$, We denote by $\mathcal{C}_\uparrow^k(\mathbb{R}^n)$ the space of the functions $f : \mathbb{R}^n \rightarrow \mathbb{R}$ which are k times differentiable and such that f and its derivatives up to order k have at most polynomial growth. For a multi-index $\alpha = (\alpha_1, \dots, \alpha_k)$ we denote $\partial_\alpha^k f = \frac{\partial^k}{\partial_{\alpha_1} \dots \partial_{\alpha_k}} f$. In the case $k = 0$, $\mathcal{C}_\uparrow^0(\mathbb{R}^n)$ designs the set of the functions $f : \mathbb{R}^n \rightarrow \mathbb{R}$ which are continuous with respect to each argument and have at most polynomial growth.

We define now the simple functionals and the simple processes.
A random variable F is called a simple functional if there exists some $n \in \mathbb{N}^*$ and some measurable function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ such that

$$F = f(V_1, \dots, V_n).$$

We denote by $S_{(n,k)}$ the space of the simple functionals such that $f \in \mathcal{C}_\uparrow^k(\mathbb{R}^n)$.

A simple process of length n is a sequence of random variables $U = (U_i)_{i \leq n}$ such that

$$U_i = u_i(V_1(\omega), \dots, V_n(\omega)).$$

We denote by $P_{(n,k)}$ the space of the simple processes of length n such that $u_i \in \mathcal{C}_\uparrow^k(\mathbb{R}^n)$, $i = 1, \dots, n$. Note that if $U \in P_{(n,k)}$ then $U_i \in S_{(n,k)}$, $i \in \mathbb{N}^*$.

We define now the differential operators which appear in the Malliavin's calculus.

□ The Malliavin derivative $D : S_{(n,1)} \rightarrow P_{(n,0)}$:
if $F = f(V_1, \dots, V_n)$, then

$$\begin{aligned} D_i F &:= \partial_i f(V_1(\omega), \dots, V_n(\omega)) = \left(\frac{\partial f}{\partial x_i} \right)(V_1(\omega), \dots, V_n(\omega)), \\ DF &:= (D_i F)_{i \leq n} \in P_{(n,0)}. \end{aligned}$$

□ The Skorohod integral $\delta : P_{(n,1)} \rightarrow S_{(n,0)}$:

$$\begin{aligned} \delta(U) &:= - \sum_{i=1}^n (D_i U_i + \theta_i(V_i) U_i) \\ &= - \sum_{i=1}^n \left(\frac{\partial u_i}{\partial x_i}(V_1, \dots, V_n) + \theta_i(V_i) u_i(V_1, \dots, V_n) \right), \end{aligned}$$

with

$$\begin{aligned}\theta_i(y) &:= \partial_y \ln(p_i)(y) = \left(\frac{p'_i}{p_i}\right)(y) & \text{if } p_i(y) > 0, \\ &:= 0 & \text{if } p_i(y) = 0.\end{aligned}$$

Remark 3.1 For every $F \in S_{(n,1)}$ and every $p \in \mathbb{N}$, one has

$$\mathbb{E} \left(\sum_{j=1}^n |D_j F|^2 \right)^p < \infty, \text{ and for every } U \in P_{(n,1)}, \mathbb{E} (|\delta(U)|^p) < \infty.$$

This follows from the fact that $\mathbb{E} |V_i|^p < \infty$, for all i (since $\lim_{x \rightarrow \pm\infty} |y|^k p_i(y) = 0$) and from the fact that f , u_i , and $\frac{\partial p_i}{p_i}$, $i = 1, \dots, n$, have at most polynomial growth.

The operator δ is the adjoint of D (this is the reason of being of θ_i):

Proposition 3.1 Let $F \in S_{(n,1)}$ and $U \in P_{(n,1)}$ then

$$E(\langle DF, U \rangle) = E(F \delta(U)), \quad (3)$$

where $\langle \cdot, \cdot \rangle$ is the scalar product in \mathbb{R}^n .

Proof. One writes

$$\begin{aligned}E(\langle DF, U \rangle) &= E \left(\sum_{i=1}^n D_i F \times U_i \right) \\ &= \sum_{i=1}^n E \left(\left(u_i \frac{\partial f}{\partial x_i} \right) (V_1, \dots, V_n) \right) \\ &= \sum_{i=1}^n \int_{\mathbb{R}^n} \left(u_i \frac{\partial f}{\partial x_i} \right) (a_1, \dots, a_n) p_1(a_1) \dots p_n(a_n) da_1 \dots da_n \\ &= - \sum_{i=1}^n \int_{\mathbb{R}^n} f(a_1, \dots, a_n) \left(\partial_i u_i + u_i \frac{p'_i}{p_i} \right) p_i(a_i) \prod_{j \neq i} p_j(a_j) da_1 \dots da_n,\end{aligned}$$

the last equality being obtained by integration by parts. We employ here the fact that for all $k \in \mathbb{N}$, $\lim_{y \rightarrow \pm\infty} |y|^k p_i(y) = 0$, $\frac{p'_i}{p_i} \in C^0_{\uparrow}(\mathbb{R}^n)$ and $f, u_i \in C^1_{\uparrow}(\mathbb{R}^n)$.

Coming back to expectations we obtain (3). ■

We introduce now the so called Ornstein Uhlenbeck operator $L = \delta(D) : S_{(n,2)} \rightarrow S_{(n,0)}$:

$$\begin{aligned} L_i F &:= D_i D_i F + \theta_i D_i F, \text{ for } i = 1, \dots, n \\ LF &:= - \sum_{i=1}^n L_i F \\ &= - \sum_{i=1}^n (\partial_i (\partial_i f)(V_1, \dots, V_n) + \theta_i(V_i) (\partial_i f)(V_1, \dots, V_n)) . \end{aligned} \quad (4)$$

Remark 3.2 Since $\theta_i, f \in \mathcal{C}_1^2(\mathbb{R}^n)$, one has $E|LF|^p < \infty$ for all p .

Suppose that $F, G \in S_{(n,2)}$ then, as an immediate consequence of the duality relation (3) one obtains

$$E(F LG) = E(\langle DF, DG \rangle) = E(G LF) . \quad (5)$$

Moreover, the standard differential calculus rules give the following chain rule.

Lemma 3.1 (i) Let $\phi \in \mathcal{C}_b^1(\mathbb{R}^d)$ and let $F = (F^1, \dots, F^d)$, $F^i \in S_{(n,1)}$. Then $\phi(F) \in S_{(n,1)}$ and

$$D\phi(F) = \sum_{k=1}^d \partial_k \phi(F) DF^k, \quad (6)$$

(ii) Let $F, G \in S_{(n,2)}$ then

$$L(FG) = F LG + G LF - 2 \langle DF, DG \rangle . \quad (7)$$

Finally, given $F = (F^1, \dots, F^d)$, $F^i \in S_{(n,1)}$, we introduce the Malliavin covariance matrix

$$\sigma_F^{ij} = \langle DF^i, DF^j \rangle = \sum_{p=1}^n (\partial_p f^i \partial_p f^j)(V_1, \dots, V_n),$$

where $F^i = f^i(V_1, \dots, V_n)$.

We are now able to state the integration by parts formula.

Theorem 3.1 *Let $F = (F^1, \dots, F^d) \in S_{(n,2)}^d$, $G \in S_{(n,1)}$. We assume that the matrix σ_F is invertible and denote $\gamma_F := \sigma_F^{-1}$.*

We also suppose that

$$E((\det \gamma_F)^4) < \infty. \quad (8)$$

Then for every smooth function $\phi : \mathbb{R}^d \rightarrow \mathbb{R}$, every $i = 1, \dots, d$

$$E(\partial_i \phi(F) G) = E(\phi(F) H^i(F, G)), \quad (9)$$

with

$$H^i(F, G) = \sum_{j=1}^d E(G \gamma_F^{ji} L F^j - \gamma_F^{ji} \langle D F^j, D G \rangle - G \langle D F^j, D \gamma_F^{ji} \rangle). \quad (10)$$

Proof. . Using the chain rule we obtain for each $j = 1, \dots, d$

$$\begin{aligned} \langle D \phi(F), D F^j \rangle &= \sum_{r=1}^n \langle D_r \phi(F), D_r F^j \rangle \\ &= \sum_{r=1}^n \sum_{i=1}^d \partial_i \phi(F) \langle D_r F^i, D_r F^j \rangle \\ &= \sum_{i=1}^d \partial_i \phi(F) \sigma_F^{ij}, \end{aligned}$$

so that $\partial_i \phi(F) = \sum_{j=1}^d \langle D \phi(F), D F^j \rangle \gamma_F^{ji}$. Moreover, using (7)

$$\langle D \phi(F), D F^j \rangle = \frac{1}{2} \left(-L(\phi(F) F^j) + \phi(F) L F^j + F^j L(\phi(F)) \right).$$

Then the duality relation gives

$$\begin{aligned}
 & E(\partial_i \phi(F) G) \\
 &= \frac{1}{2} E \left(\left(G \left(\sum_{j=1}^d -L(\phi(F) F^j) + \phi(F) L F^j + F^j L(\phi(F)) \right) \right) \gamma_F^{ji} \right) \\
 &= \sum_{j=1}^d \frac{1}{2} E \left(\phi(F) \left(-F^j L(\gamma_F^{ji} G) + G \gamma_F^{ji} L F^j + L(G F^j \gamma_F^{ji}) \right) \right) \\
 &= \sum_{j=1}^d E \left(\phi(F) \left(G \gamma_F^{ji} L F^j - \langle D F^j, D(G \gamma_F^{ji}) \rangle \right) \right),
 \end{aligned}$$

and the proof is completed. ■

4 Integration by parts with respect to the jump amplitudes

In this section we use the integration by parts formula (9) with respect to $\Delta_i \sim \rho(a) da$, $i \in \mathbb{N}^*$, which are the amplitudes of the jumps of $(N_t)_{t \geq 0}$, solution of the SDE

$$S_t = \beta + \sum_{i=1}^{J_t} c(T_i, \Delta_i, S_{T_i-}) + \int_0^t b(r, S_r) dr. \quad (11)$$

We work under the following hypothesis.

Hypothesis 4.1 ρ is continuously differentiable, $\frac{\rho'}{\rho}$ has at most polynomial growth on \mathbb{R} and for all $k \in \mathbb{N}$, $\lim_{y \rightarrow \pm\infty} |y|^k \rho(y) = 0$.

Hypothesis 4.2 The functions $(a, x) \rightarrow c(t, a, x)$ and $x \rightarrow b(t, x)$ are twice continuously differentiable and there exists a constant $K > 0$ and $\alpha \in \mathbb{N}$ such

that:

$$\begin{aligned}
i) \quad & |c(t, a, x)| \leq K (1 + |a|)^\alpha (1 + |x|) \\
& |b(t, x)| \leq K (1 + |x|) \\
ii) \quad & |\partial_x c(t, a, x)| + |\partial_x^2 c(t, a, x)| + |\partial_a c(t, a, x)| \leq K (1 + |a|)^\alpha \\
& |\partial_x b(t, x)| + |\partial_x^2 b(t, x)| \leq K
\end{aligned}$$

Hypothesis 4.3

$$|\partial_a c(t, a, x)| \geq \eta > 0.$$

4.1 The deterministic equation

We introduce the following deterministic equation.

We fix some $0 < u_1 < \dots < u_n < T$, and we denote $u = (u_1, \dots, u_n)$. We also fix $a = (a_1, \dots, a_n) \in \mathbb{R}^n$. To u and a , we associate the equation

$$s_t = \beta + \sum_{i=1}^{J_t(u)} c(u_i, a_i, s_{u_i^-}) + \int_0^t b(r, s_r) dr, \quad 0 \leq t \leq T, \quad (12)$$

where $J_t(u) = k$ if $u_k \leq t < u_{k+1}$.

We use this deterministic equation in order to express S_t as a simple functional: on $\{J_t = n\}$ one has

$$\begin{aligned}
S_t &= s_t(T_1, \dots, T_n, \Delta_1, \dots, \Delta_n), \\
\partial_{\Delta_i} S_t &= \partial_{a_i} s_t(T_1, \dots, T_n, \Delta_1, \dots, \Delta_n), \\
\partial_{\Delta_j, \Delta_i}^2 S_t &= \partial_{a_j, a_i}^2 s_t(T_1, \dots, T_n, \Delta_1, \dots, \Delta_n), \\
\partial_\beta S_t &= \partial_\beta s_t(T_1, \dots, T_n, \Delta_1, \dots, \Delta_n).
\end{aligned}$$

So in order to compute the derivatives of S_t , we have to compute those of s_t . Under hypothesis 4.2, we may perform these computations by differentiating with respect to a_j , $j = 1, \dots, n$ in (12). We obtain

- For $i \leq J_t(u)$, $\partial_{a_i} s_t$ satisfies the equation

$$\begin{aligned} \partial_{a_i} s_t = \partial_a c(u_i, a_i, s_{u_i}^-) + \sum_{j=i+1}^{J_t(u)} \partial_x c(u_j, a_j, s_{u_j}^-) \partial_{a_i} s_{u_j}^- \\ + \int_{u_i}^t \partial_x b(r, s_r) \partial_{a_i} s_r dr. \end{aligned} \quad (13)$$

•

$$\begin{aligned} \partial_{a_i}^2 s_t = \partial_a^2 c(u_i, a_i, s_{u_i}^-) + \sum_{j=i+1}^{J_t(u)} \partial_x^2 c(u_j, a_j, s_{u_j}^-) (\partial_{a_i} s_{u_j}^-)^2 \\ + \sum_{j=i+1}^{J_t(u)} \partial_x c(u_j, a_j, s_{u_j}^-) \partial_{a_i}^2 s_{u_j}^- + \int_{u_i}^t \partial_x b(r, s_r) \partial_{a_i}^2 s_r dr \\ + \int_{u_i}^t \partial_x^2 b(r, s_r) (\partial_{a_i} s_r)^2 dr. \end{aligned} \quad (14)$$

- For $i < j$

$$\begin{aligned} \partial_{a_j, a_i}^2 s_t = \partial_{a, x}^2 c(u_j, a_j, s_{u_j}^-) + \sum_{k=j+1}^{J_t(u)} \partial_x^2 c(u_k, a_k, s_{u_k}^-) \partial_{a_i} s_{u_k}^- \partial_{a_j} s_{u_k}^- \\ + \sum_{k=j+1}^{J_t(u)} \partial_x c(u_k, a_k, s_{u_k}^-) \partial_{a_j, a_i}^2 s_{u_k}^- + \int_{u_j}^t \partial_x b(r, s_r) \partial_{a_j, a_i}^2 s_r dr \\ + \int_{u_k}^t \partial_x^2 b(r, s_r) \partial_{a_i} s_r \partial_{a_j} s_r dr, \end{aligned} \quad (15)$$

and for $i > j$, we obtain $\partial_{a_j, a_i}^2 s_t$ by symmetry.

- $\partial_\beta s_t$ is computed by

$$\partial_\beta s_t = 1 + \sum_{i=1}^{J_t(u)} \partial_x c(u_i, a_i, s_{u_i}^-) \partial_\beta s_{u_i}^- + \int_0^t \partial_x b(r, s_r) \partial_\beta s_r dr. \quad (16)$$

Lemma 4.1 (i) The function $(a_1, \dots, a_n) \rightarrow s_t(u_1, \dots, u_n, a_1, \dots, a_n) \in \mathcal{C}_\uparrow^2(\mathbb{R}^n)$, and $(a_1, \dots, a_n) \rightarrow \partial_\beta s_t(u_1, \dots, u_n, a_1, \dots, a_n) \in \mathcal{C}_\uparrow^1(\mathbb{R}^n)$.
(ii) $|\partial_{a_n} s_t|^2 \geq \eta^2 e^{-TK} > 0$.

Proof. (i) Proceed by recurrence using hypothesis 4.2 in (12), (13), (14), (15), and in (16).

(ii) Using (13), $\partial_{a_n} s_T = \partial_a c(t_n, a_n, s_{t_n}^-) + \int_{t_n}^T \partial_x b(r, s_r) \partial_{a_n} s_r dr$, so

$$|\partial_{a_n} s_T| = |\partial_a c(t_n, a_n, s_{t_n}^-)| \exp \left(\int_{t_n}^T \partial_x b(r, s_r) dr \right) \geq \eta e^{-TK}.$$

■

4.2 Integration by parts formula

We come back to the problem in section 1 and we deal with the equation (9). We write

$$\begin{aligned} E \left(\phi'(S_T) \frac{\partial S_T}{\partial \beta} \mathbf{1}_{J_T=n} \right) &= E \left[E \left(\phi'(S_T) \frac{\partial S_T}{\partial \beta} | \sigma(T_i, i \in \mathbb{N}) \right) \mathbf{1}_{J_T=n} \right] \\ &= E (h_n(T_1, \dots, T_n) \mathbf{1}_{J_T=n}), \end{aligned}$$

with

$$\begin{aligned} &h_n(t_1, \dots, t_n) \\ &= E \left[\phi'(s_T(t_1, \dots, t_n, \Delta_1, \dots, \Delta_n)) \frac{\partial s_T}{\partial \beta}(t_1, \dots, t_n, \Delta_1, \dots, \Delta_n) \right]. \end{aligned}$$

We fix $n \geq 1$ and t_1, \dots, t_n and we apply Theorem 3.1 with

$F = s_T(t_1, \dots, t_n, \Delta_1, \dots, \Delta_n)$ and $G = \partial_\beta s_T(t_1, \dots, t_n, \Delta_1, \dots, \Delta_n)$.

By Lemma 4.1, we know that $F \in S_{(n,2)}$ and $G \in S_{(n,1)}$, so we have to verify that σ_F is invertible and that (8) holds true and then we compute

$$H_n(F, G) = G \gamma_F L F - \gamma_F \langle DF, DG \rangle - G \langle DF, D\gamma_F \rangle. \quad (17)$$

We have

$$\begin{aligned}\sigma_F &= \sum_{i=1}^n |\partial_{a_i} s_T(t_1, \dots, t_n, \Delta_1, \dots, \Delta_n)|^2 \\ &\geq |\partial_{a_n} s_T(t_1, \dots, t_n, \Delta_1, \dots, \Delta_n)|^2 \geq \eta^2 e^{-2TK},\end{aligned}$$

So we can use Theorem 3.1 and we obtain the integration by parts formula (9). For the computation of the terms involved in the right hand side of (17),

- we use (13) for DF and γ_F , (16) for G , (13) and (14) for LF , (13), (14) and (15) for $D\gamma_F$,
- we differentiate with respect to a_j in (16) to obtain DG .

5 Merton Process and Euler scheme

In this section we deal with the Merton model:

$$S_t = \beta + \sum_{i=1}^{J_t} c(T_i, \Delta_i, S_{T_i}^-) + \int_0^t b(u, S_u) du + \int_0^t \sigma(u, S_u) dW_u,$$

where ρ , b and c satisfy the previous hypothesis 4.1, 4.2 and 4.3. Moreover, we assume

Hypothesis 5.1 *The function $x \rightarrow \sigma(u, x)$ is twice continuously differentiable and there exists a constant $C > 0$ such that:*

$$\begin{aligned}i) \quad & |\sigma(u, x)| \leq C(1 + |x|), \\ ii) \quad & |\partial_x \sigma(u, x)| + |\partial_x^2 \sigma(u, x)| \leq C, \\ iii) \quad & |\sigma(u, x)| \geq \eta.\end{aligned}$$

We will present two alternative calculus for this model. The first one is based on the Brownian motion only and the second one is based on both the Brownian motion and the jump amplitudes. Suppose that the jump times $T_1 < \dots < T_n$ are given (in the Monte-Carlo algorithm this means that we have already simulated T_1, \dots, T_n). We include them in the discretization grid: so we consider

a time grid $0 = t_0 < t_1 < \dots < t_m < \dots < t_M = T$ and we assume that $T_i = t_{m_i}$, $i = 1, \dots, n$ for some $m_1 < \dots < m_n$. For $t > 0$, we denote $m(t) = m$ if $t_m \leq t < t_{m+1}$. Then the corresponding Euler scheme is given by

$$\begin{aligned} \hat{S}_t = \beta + \sum_{i=1}^{J_t} c(T_i, \Delta_i, \hat{S}_{T_{i-}}) + \sum_{k=0}^{m(t)-1} \sigma(t_k, \hat{S}_{t_k}) (W_{t_{k+1}} - W_{t_k}) \\ + \sum_{k=0}^{m(t)-1} b(t_k, \hat{S}_{t_k}) (t_{k+1} - t_k). \end{aligned}$$

This corresponds to the deterministic equation :

$$\hat{s}_t = \beta + \sum_{i=1}^{J_t} c(u_i, a_i, \hat{s}_{u_i^-}) + \sum_{k=0}^{m(t)-1} \sigma(t_k, \hat{s}_{t_k}) \Delta_k w + \sum_{k=0}^{m(t)-1} b(t_k, \hat{s}_{t_k}) (t_{k+1} - t_k), \quad (18)$$

where we have denoted by $\Delta_k w = w_{t_{k+1}} - w_{t_k}$. So on $\{J_t = k\}$, one has

$$\hat{S}_t = \hat{s}_t(T_1, \dots, T_k, \Delta_1, \dots, \Delta_k, \Delta_0 W, \dots, \Delta_{m(t)-1} W),$$

where $\Delta_k W = W_{t_{k+1}} - W_{t_k}$. Moreover we have :

$$\begin{aligned} \partial_{\Delta_i} \hat{S}_t &= \partial_{a_i} \hat{s}_t(T_1, \dots, T_k, \Delta_1, \dots, \Delta_k, \Delta_0 W, \dots, \Delta_{m(t)-1} W), \\ \partial_{\Delta_j, \Delta_i}^2 \hat{S}_t &= \partial_{a_j, a_i}^2 \hat{s}_t(T_1, \dots, T_k, \Delta_1, \dots, \Delta_k, \Delta_0 W, \dots, \Delta_{m(t)-1} W), \\ \partial_{\beta} \hat{S}_t &= \partial_{\beta} \hat{s}_t(T_1, \dots, T_k, \Delta_1, \dots, \Delta_k, \Delta_0 W, \dots, \Delta_{m(t)-1} W), \\ \partial_{\Delta_k W} \hat{S}_t &= \partial_{\Delta_k w} \hat{s}_t(T_1, \dots, T_k, \Delta_1, \dots, \Delta_k, \Delta_0 W, \dots, \Delta_{m(t)-1} W). \end{aligned}$$

The first derivatives of \hat{s}_t satisfy the following equations :

$$\begin{aligned} \partial_{a_i} \hat{s}_t = \partial_a c(u_i, a_i, \hat{s}_{u_i^-}) + \sum_{k=i+1}^{J_t} \partial_x c(u_k, a_k, \hat{s}_{u_k^-}) \partial_{a_i} \hat{s}_{u_k^-} \\ + \sum_{k=i}^{m(t)-1} \partial_x b(t_k, \hat{s}_{t_k}) \partial_{a_i} \hat{s}_{t_k} \delta_k + \sum_{k=i}^{m(t)-1} \partial_x c(t_k, \hat{s}_{t_k}) \partial_{a_i} \hat{s}_{t_k} \Delta_k w, \quad (19) \end{aligned}$$

$$\begin{aligned} \partial_{\Delta_i w} \hat{s}_t &= \sigma(t_i, \hat{s}_{t_i}) + \sum_{k=1}^{J_t} \partial_x c(u_k, a_k, \hat{s}_{u_k}^-) \partial_{\Delta_i w} \hat{s}_{u_k}^- \\ &+ \sum_{k=i}^{m(t)-1} \partial_x \sigma(t_k, \hat{s}_{t_k}) \partial_{\Delta_i w} \hat{s}_{t_k} \Delta_k W + \sum_{k=i}^{m(t)-1} \partial_x b(t_k, \hat{s}_{t_k}) \partial_{\Delta_i w} \hat{s}_{t_k} \delta_k, \end{aligned} \quad (20)$$

where $\delta_k = t_{k+1} - t_k$.

For the derivatives of higher order one may derive similar equations. Now we have the choice of using the integration by parts formula from the section 2 using $\Delta_i W$ or both of Δ_i and $\Delta_i W$. And in each case we have different forms for the differential operators. For example on the set $\{J_t = k\}$

$$\sigma_t^{J,W} := \sum_{i=1}^k |\partial_{a_i} \hat{s}_t|^2 + \sum_{i=0}^{m(t)-1} |\partial_{\Delta_i w} \hat{s}_t|^2, \quad (21)$$

$$\sigma_t^J := \sum_{i=1}^k |\partial_{a_i} \hat{s}_t|^2, \quad (22)$$

$$\sigma_t^W := \sum_{i=0}^{m(t)-1} |\partial_{\Delta_i w} \hat{s}_t|^2. \quad (23)$$

$$(24)$$

The other differential operators will change in a similar way. Note that $\Delta_i W \sim \mathcal{N}(0, t_{i+1} - t_i)$ so that the corresponding L_i^W is given by

$$L_i^W \hat{s}_t = \partial_{\Delta_i w}^2 \hat{s}_t + \theta_i^W \partial_{\Delta_i w} \hat{s}_t, \text{ with } \theta_i^W = -\frac{\Delta_i W}{t_{i+1} - t_i}.$$

Then the Ornstein-Uhlenbeck operator will be

$$\begin{aligned} L^W \hat{S}_t &= - \sum_{i=0}^{m(t)-1} L_i^W \hat{s}_t, \\ L^J \hat{S}_t &= - \sum_{i=1}^{J_t} \partial_{\Delta_i}^2 \hat{s}_t + \theta_i^J \partial_{\Delta_i} \hat{s}_t, \\ L^{J,W} \hat{S}_t &= L^J \hat{S}_t + L^W \hat{S}_t. \end{aligned}$$

Notice that if $m = m(t)$

$$\sigma_t^{J,W} \geq \sigma_t^W \geq |\partial_{\Delta_{m-1}W} \hat{s}_t|^2 = |\sigma(t_{m-1}, \hat{s}_{t_{m-1}})|^2 \geq \eta^2 > 0.$$

This allows us to use the integration by parts formula corresponding to the Brownian motion only, or to both Brownian motion and jump amplitudes (in the first case one obtains the same calculus as in [6] [13]). It is more delicate to prove the non degeneracy condition if we use only the jumps.

6 Numerical results

We compute here the Delta of two European options (call option with payoff $\phi(x) = (x - K)_+$ and digital option with payoff $\phi(x) = \mathbf{1}_{x \geq K}$) with underlying $(S_t)_{t \geq 0}$ following a jump type diffusion model.

Let N be a compound Poisson process with jump intensity λ . For all $i \in \mathbb{N}^*$, we denote Δ_i the jump amplitude of N at the jump time T_i . We suppose that $\Delta_i \sim \mathcal{N}(0, 1)$, $i \geq 1$.

We deal with two different jump diffusion models. The first one is motivated by Vasicek's model for interest rates (but we consider a jump process instead of a Brownian motion):

$$S_t = S_0 - \int_0^t r(S_u - \alpha) du + \sigma \sum_{i=1}^{J_t} \Delta_i. \quad (25)$$

The second one is of Black-Scholes type:

$$S_t = S_0 + \int_0^t r S_u du + \sigma \sum_{i=1}^{J_t} S_{T_i^-} \Delta_i, \quad (26)$$

We compare the different Monte Carlo estimators of the Delta, obtained by using

- Finite difference method
- Malliavin calculus
- localized Malliavin calculus.

6.1 Finite Difference method

Arbitrage theory gives an expression for the price $u(,)$ of an option, with underlying S and payoff ϕ , as the following expected value :

$$u(0, S_0) = \mathbb{E}[\phi(S_T)|S_0] .$$

To compute the Delta, the finite difference method makes a differentiation using the Taylor expansion of the price with respect to S_0 . Indeed, we shift S_0 with ϵ and compute the new price $u(0, S_0 + \epsilon)$, then the first term of the Taylor expansion of the price around S_0 is given by:

$$\frac{\partial u(0, S_0)}{\partial S_0} \simeq \frac{u(0, S_0 + \epsilon) - u(0, S_0 - \epsilon)}{2\epsilon} .$$

We choose the symmetric estimator and we use the same simulated paths in the two "shifted expectation" in order to reduce the variance.

6.2 Malliavin estimator

We recall that we have computed the Delta estimator using Malliavin integration by parts formula (see section 2):

$$\mathbb{E} \left(\phi'(S_T) \frac{\partial S_T}{\partial \beta} \right) \simeq \frac{1}{M} \sum_{i=1}^M \phi(S_T^i) H_{J_T^i}^i(S_T^i, \partial_\beta S_T^i) 1_{J_T^i \geq 1} + \phi'(S_T^i) \partial_\beta S_T^i 1_{J_T^i = 0} .$$

We compute here the Malliavin weights $H_{J_T^i}^i(S_T^i, \partial_\beta S_T^i)$.

- We first study the diffusion process defined by (25). We have an explicit expression of S_T on $\{J_T = n\}$:

$$S_T = S_0 e^{-rT} + \alpha (1 - e^{-rT}) + \sigma \sum_{j=1}^n \Delta_j e^{-r(T-T_j)} . \quad (27)$$

In order to compute the Malliavin derivatives of S_T involved in the weight H_{J_T} , we differentiate with respect to the jump amplitudes in (27). Thus, for

all $1 \leq i \leq n$, one has

$$\begin{aligned} D_i S_T &= \sigma e^{-r(T-T_i)} \\ D_{ii}^2 S_T &= 0 \\ Y_T &:= \frac{\partial S_T}{\partial S_0} = e^{-rT} \\ D_i Y_T &= 0, \end{aligned}$$

and the covariance matrix is defined by :

$$\sigma_T = \sum_{j=1}^n |D_j S_T|^2 = \sigma^2 \sum_{j=1}^n e^{-2r(T-T_j)}.$$

Then $\gamma_T = \frac{1}{\sigma_T} \Rightarrow D_i \gamma_T = 0$, for all $1 \leq i \leq n$.

Since $\frac{\partial \ln \rho(\Delta)}{\partial \Delta} = -\Delta$, one has

$$L S_T = - \sum_{j=1}^n D_j S_T \frac{\partial \ln \rho(\Delta_j)}{\partial \Delta_j} = \sum_{j=1}^n \sigma e^{-r(T-T_j)} \Delta_j.$$

Finally, using equation (10), on $(J_T = n)$

$$H_n(S_T, Y_T) = \frac{\sum_{j=1}^n e^{rT_j} \Delta_j}{\sigma \sum_{j=1}^n e^{2rT_j}}. \quad (28)$$

• We study the jump diffusion process defined by (26).
On $(J_T = n)$, we have

$$S_T = S_0 e^{rT} \prod_{j=1}^n (1 + \sigma \Delta_j),$$

so, for all $1 \leq i \leq n$,

$$D_i S_T = \frac{\sigma S_T}{1 + \sigma \Delta_i} = \sigma \prod_{j=1, j \neq i}^n (1 + \sigma \Delta_j).$$

Notice that if $(1 + \sigma \Delta_i) = 0$, then $S_T = 0$. So we employ the convention $\frac{0}{0} = 0$. This is just a matter of notation.

$$D_i S_T = \frac{\sigma S_T}{1 + \sigma \Delta_i} \text{ for all } i \in [1..n].$$

Let us define

$$A_\sigma = \sum_{j=1}^n \frac{1}{(1 + \sigma \Delta_j)^2} \quad (29)$$

$$B_\sigma = \sum_{j=1}^n \frac{\Delta_j}{(1 + \sigma \Delta_j)} \quad (30)$$

$$C_\sigma = \sum_{j=1}^n \frac{1}{(1 + \sigma \Delta_j)^4}, \quad (31)$$

then we get, for all $1 \leq i \leq n$

$$\begin{aligned} D_{ii}^2 S_T &= 0 \\ Y_T &= \frac{S_T}{S_0} \\ D_i Y_T &= \frac{\sigma S_T}{S_0 (1 + \sigma \Delta_i)} \\ \sigma_T &= \sigma^2 S_T^2 \sum_{j=1}^n \frac{1}{(1 + \sigma \Delta_j)^2} = \sigma^2 S_T^2 A_\sigma \\ D_i \sigma_T &= \left(\frac{2 \sigma^3 S_T^2}{1 + \sigma \Delta_i} \right) \left(A_\sigma - \frac{1}{(1 + \sigma \Delta_i)^2} \right) \\ D_i \gamma_T &= -\frac{D_i \sigma_T}{\sigma_T^2}. \end{aligned}$$

Hence,

$$H_n(S_T, Y_T) = \frac{B_\sigma}{\sigma S_0 A_\sigma} + \frac{1}{S_0} - \frac{2 C_\sigma}{S_0 A_\sigma^2}. \quad (32)$$

After simulating trajectories of the specified diffusion, we can compute the Malliavin weights by (28) and (32). Note that we only need to know the amplitudes and the times of the jumps to state a Monte-Carlo estimator of the Delta.

6.3 Merton Process

Here we focus on the Merton model (2):

$$S_t = S_0 + \int_0^t r S_u du + \int_0^t \sigma S_u dW_u + \mu \sum_{i=1}^{J_t} S_{T_i^-} \Delta_i, \quad (33)$$

where W is a Brownian motion independent on the compound Poisson process N . We suppose that the jump amplitudes Δ_i are independent, identically and gaussian distributed. We have the explicit solution:

$$S_T = S_0 \exp \left(\left(r - \frac{\sigma^2}{2} \right) T + \sigma W_T \right) \prod_{j=1}^{J_T} (1 + \mu \Delta_j), \quad (34)$$

so we do not need to use Euler's scheme.

On $\{J_T = n\}$, $n \in \mathbb{N}^*$, the source of randomness is $H = (\Delta_1, \dots, \Delta_n, W_T)$. For all $i \in \{1, \dots, n\}$,

$$D_i^N(S_T) := \frac{\partial S_T}{\partial \Delta_i} = \frac{\mu S_T}{1 + \mu \Delta_i} \quad (35)$$

$$D^W(S_T) := \frac{\partial S_T}{\partial W_T} = \sigma S_T. \quad (36)$$

Then we can compute all the terms involved in the Malliavin weight.

$$\begin{aligned} D_i^N(D_i^N S_T) &= 0 \\ D^W(D^W S_T) &= \sigma^2 S_T \\ Y_T &= \frac{S_T}{S_0} \\ D_i^N(Y_T) &= \frac{\mu S_T}{S_0(1 + \mu \Delta_i)} \\ D^W(Y_T) &= \sigma Y_T. \end{aligned}$$

The covariance matrix is

$$\sigma_T = |D^W(S_T)|^2 + \sum_{i=1}^n |D_i^N(S_T)|^2 = \mu^2 S_T^2 \sum_{j=1}^n \frac{1}{(1 + \mu \Delta_j)^2} + \sigma^2 S_T^2.$$

Straightforward computations give

$$\begin{aligned} D_i^N(\sigma_T) &= \frac{2\mu^3 S_T^2}{(1 + \mu \Delta_i)} (A_\mu - \frac{1}{(1 + \mu \Delta_i)^2}) + \frac{2\sigma^2 \mu S_T^2}{1 + \mu \Delta_i} \\ D^W(\sigma_T) &= 2\sigma \mu^2 S_T^2 A_\mu + 2\sigma^3 S_T^2 \\ D_i^N(\gamma_T) &= -\frac{D_i^N(\sigma_T)}{\sigma_T^2} \\ D^W(\gamma_T) &= -\frac{D^W(\sigma_T)}{\sigma_T^2}, \end{aligned}$$

where A_μ is given by (29).

We put these terms together to get the Malliavin weight:

$$H = \frac{\mu B_\mu + \sigma \frac{W_T}{T} - \sigma^2}{S_0(\mu^2 A_\mu + \sigma^2)} + \frac{1}{S_0} - \frac{2\mu^4 C_\mu}{S_0(\mu^2 A_\mu + \sigma^2)^2}, \quad (37)$$

where B_μ and C_μ are defined by (30) and (31) respectively.

Recently in [6] and [13] one computes the Delta of an European option using the Malliavin calculus with respect to the Brownian motion only. Notice that if we use our integration by parts formula, but we take into account

just the derivatives with respect to the Brownian motion, we find $H = \frac{W_T}{S_0 \sigma T}$, which is exactly the weight obtained in [13] (as well as in Black-Scholes). So the difference between our algorithm and the one in [13] consists in the fact that the weight H contains a supplementary part corresponding to the derivatives with respect to the jump amplitudes. In figure 2, we compare the two algorithms. Moreover, in table [1], we give the quotient between the empirical variances of the two algorithms. It turns out that the variance of the Brownian-jump algorithm (presented here) is smaller than the variance of the pure Brownian algorithm (presented in [13]). Moreover, the difference becomes more important as the number of jumps up to T becomes larger: this occurs when the maturity T is larger or when the intensity λ of the Poisson measure is larger. We conclude that more noise one uses in the integration by parts formula, better the algorithm works (there is no theoretical result in this sense, but only numerical evidence).

$T \setminus \lambda$	1	4	8	12
1	2,15	7,27	19,88	16,43
2	1,72	12,17	22,12	36,44
3	2,94	7,15	24,30	35,58

Table 1: $\frac{\text{variance Brownian}}{\text{variance Brownian-Jump}}$ for Digital delta with different maturity and jump intensity

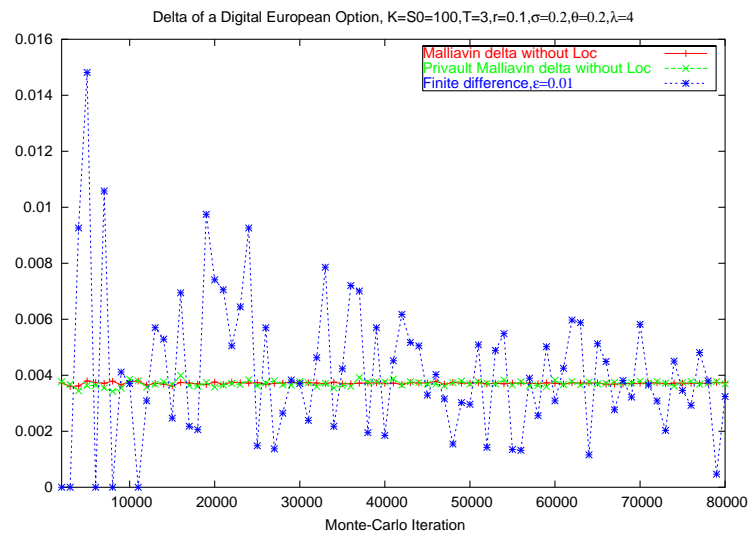


Figure 1: Delta of Digital option for a Merton Process

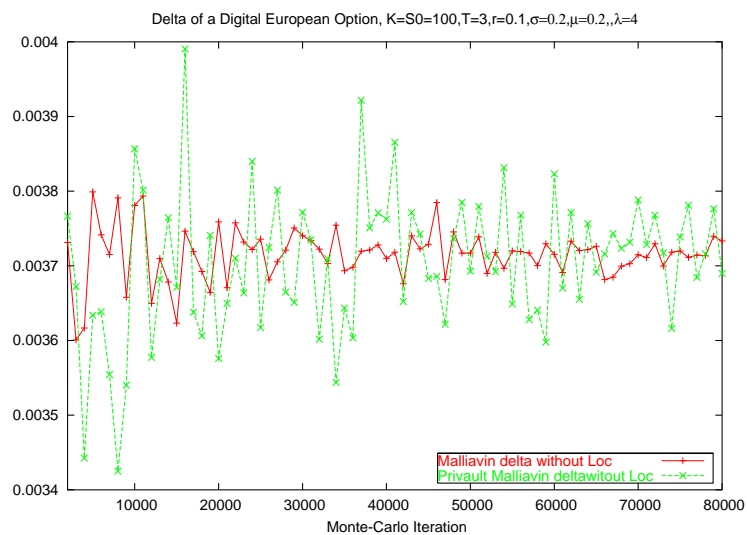


Figure 2: Zoom of figure 1

6.4 Localization method

In [9] one proposes a localization method which reduces the variance of the estimators. We use here the same method and we see that the results are significantly improved. Actually, the computation of Greeks using Malliavin calculus leads to terms such as $\phi(S_T) H_n(S_T, \frac{\partial S_T}{\partial \beta})$, which could have a large variance. It's possible to avoid this problem by introducing a localizing function which vanishes out of an interval $[K - \delta, K + \delta]$.

6.5 figures and comments

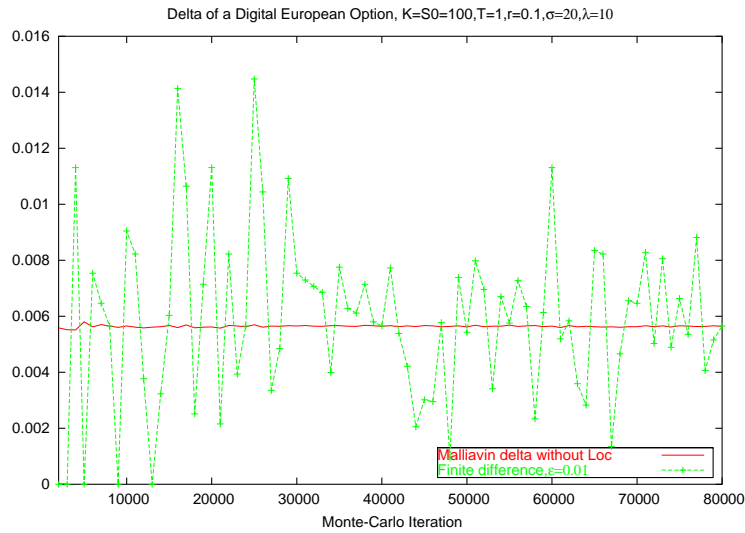


Figure 3: Delta of an European digital option using Malliavin calculus and finite Difference Method. Equation (1).

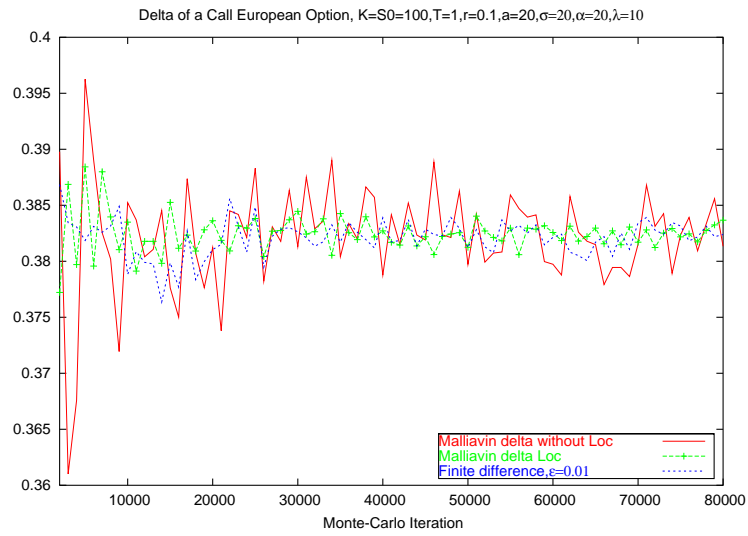


Figure 4: Delta of an European call option using Malliavin calculus and finite Difference Method. Equation (1).

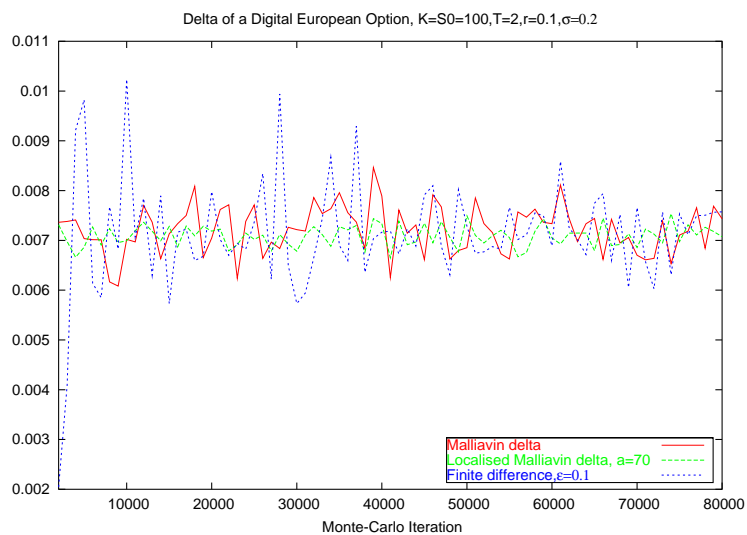


Figure 5: Delta of an European digital option using Malliavin calculus and finite Difference Method. Equation (2).

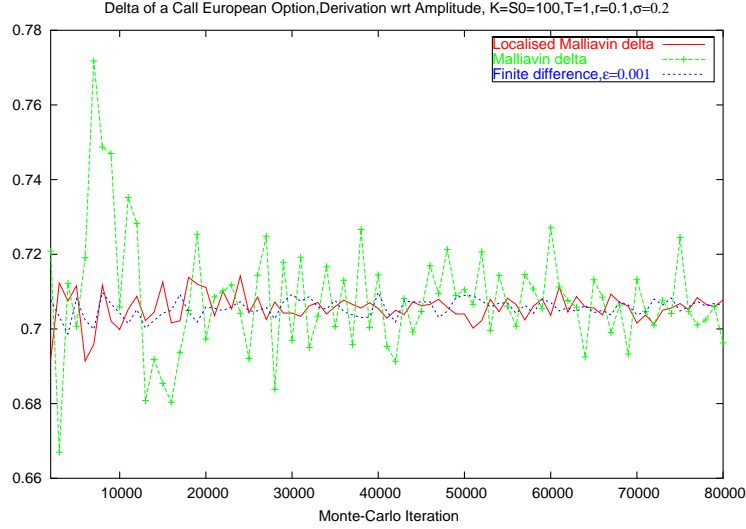


Figure 6: Delta of an European call option using Malliavin calculus and finite Difference Method. Equation (2).

These figures confirm that we can numerically compute the Greeks for European options with a pure jump underlying process. We obtain numerical results similar to those in the Wiener case ([9] and [8]).

On the one hand, for European Call options, the Malliavin estimator has more variance than the finite difference one (see figures 4 and 6): the finite difference method approximates the first derivative of the payoff, whereas the Malliavin estimator contains a weight independent on the payoff, which may increase the variance. Using a localization technique permits to reduce the variance of the Malliavin estimator and to make it closer to the finite difference one.

On the other hand, the Malliavin estimator of a Digital option has less variance than the finite difference one (see figures 3 and 5) and so does not need to be localized: in this case, the first derivative of the payoff is a Dirac and, contrary to the finite difference method, the Malliavin calculus permits to avoid this strong discontinuity.

Finally, notice that for both call and digital options, the finite difference method requires to simulate twice more samples of the asset than the Malliavin

method does : the finite difference method uses the samples starting from S_0 and those starting from $S_0 + \epsilon$. So the Malliavin method is less time consuming.

7 Conclusion

In order to develop a Malliavin calculus for jump processes, we worked here on a space of simple functionals of a finite set of random variables representing the source of randomness. First, we applied the Malliavin calculus to pure jump processes, by differentiating with respect to the jump amplitudes. Second, in the case of Merton process, we used the total randomness coming from the jump amplitudes and the Brownian increments. Then, under some standard assumptions, we have set up an integration by parts formula, which permits to replace the derivative of the payoff by a Malliavin weight in the computation of the Greeks.

The Numerical experiments show that using Malliavin approach becomes extremely efficient for a discontinuous payoff. Some specific technics can be used to reduce the variance of the Malliavin estimator.

For the Merton process, we compare our Delta estimator with the one where one omits the presence of the jump process (that is we don't differentiate with respect to the jump amplitudes). The results show that it is better to use the two sources of randomness especially when we have more jumps.

References

- [1] F. Biagini, B. Øksendal, A. Sulem, and N. Wallner. An introduction to white noise and Malliavin calculus for fractional Brownian motion. *To appear in Proc. Royal Soc. London.*, page 35 pages, 2003.
- [2] K. Bichteler, J. B. Gravereaux, and J. Jacod. *Malliavin calculus for processes with jumps*. Gordon and Breach, 1987.
- [3] N. Bouleau. *Error Calculus For Finance and Physics, The Language of Dirichlet Forms*. De Gruyter, 2003.

- [4] Eric A. Carlen and Étienne Pardoux. Differential calculus and integration by parts on Poisson space. *Kluwer ed, Stochastics, algebra and analysis in classical and quantum dynamics.*, pages 63–73, 1990.
- [5] R. Cont and P. Tankov. *Financial modelling with Jump Processes*. Chapman , Hall / CRC Press, 2003.
- [6] M. H. A. Davis and M. Johansson. Malliavin Monte Carlo Greeks for jump diffusions. *Preprint submitted to Stochastic Processes and their Applications.*, 2004.
- [7] B. Øksendal. An introduction to Malliavin calculus with applications to economics. Lecture notes, Norwegian School of Economics and Business Administration, Norway, 1996.
- [8] E. Fournié, J. M. Lasry, J. Lebouchoux, and P. L. Lions. Applications of Malliavin Calculus to Monte Carlo Methods in Finance II. *Finance Stoch.*, 2:73–88, 2001.
- [9] E. Fournié, J. M. Lasry, J. Lebouchoux, P. L. Lions, and N. Touzi. Applications of Malliavin calculus to Monte Carlo Methods in Finance. *Finance Stoch.*, 5(2):201–236, 1999.
- [10] Y. El Khatib and N. Privault. Pricing and hedging derivatives securities in markets with uncertain volatilities. *Finance Stoch.*
- [11] D. Nualart and J. Vives. Anticipative calculus for the Poisson process based on the Fock space. *sem Proba. XXIV, Lect. Notes in Math.*, 1426:154–165, Springer (1990).
- [12] G. Di Nunno, B. Øksendal, and F. Proske. Malliavin calculus for Lévy processes. *To appear in Proc. Royal Soc. London.*, Manuscript, 2003.
- [13] N. Privault. Malliavin Monte Carlo Greeks for jump diffusions. *Preprint.*, 2004.
- [14] J. Vives, J. A. León, F. Utzet, and J. L. Solé. On Lévy processes, Malliavin calculus and market models with jumps. *Finance Stoch.*, 6(2):197–225, 2002.



Unité de recherche INRIA Rocquencourt
Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)
